MIDTERM – ENPM665 0101
GROUP NUMBER - 16

MEMBERS NAME AND DIRECTORY ID:

1. **Bhavik Chirag Shah [bshah007]**
2. **Nikita Ganapati Patil [nikspat]**
3. **Yogi Bipinkumar Makadiya [yogim]**

Section 1: Introduction

In an age where the healthcare industry is becoming increasingly reliant on digital solutions, the protection of sensitive patient data is of paramount importance. The digitization of medical records, personal information, and billing details has brought about numerous advantages in terms of accessibility and efficiency, but it has also raised significant security concerns. The cloud infrastructure supporting these healthcare applications plays a critical role in ensuring the confidentiality, integrity, and availability of this sensitive data.

This vulnerability assessment report delves into the security posture of a healthcare company's cloud infrastructure. We have the task of identifying and mitigating potential security risks within their cloud environment, which houses a comprehensive healthcare application stack. This stack comprises web servers, application servers, and database servers, all of which are pivotal to the smooth functioning of healthcare services. In the current scenario, the asset list of the infrastructure looks like the following:

1. *EC2 Instances:* These are deployed to host the front-end and back-end of the web application stack, making them crucial components of healthcare service delivery.
2. *VPC*: The Virtual Private Cloud (VPC) is currently exposed to the entire internet with no policy assigned to it, posing a significant security risk to the healthcare company.
3. *RDS*: The Relational Database Service (RDS) is the database used to store and manage sensitive patient data, conduct telemedicine consultations, and host critical healthcare applications.
4. *S3*: The Simple Storage Service (S3) is utilized to store product information, user details, medical history, and items for web applications, making it a key repository for sensitive data.

5. *IAM Policies*: These policies are designed for developers, cloud admin, and guests but have been loosely configured with improper access controls and excessive permissions, which could lead to unauthorized access and potential breaches.

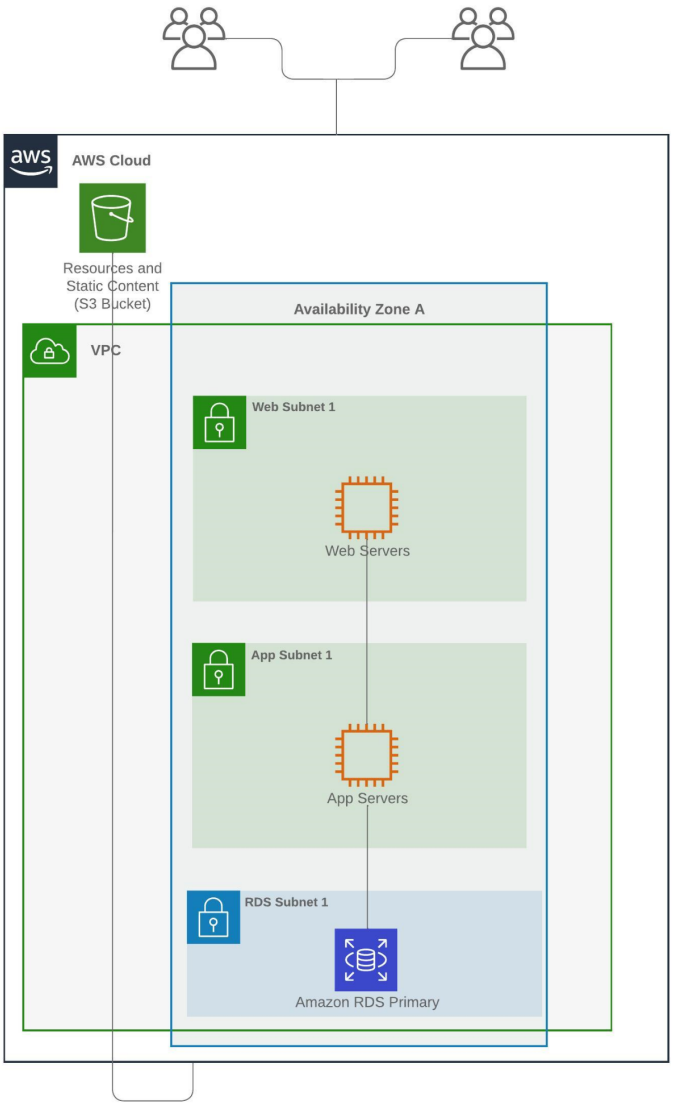


Figure 1: Current architecture of the healthcare application.

The scope of our assessment covers various facets of cloud security, including identity and access management, data encryption, network security, and vulnerability management. It is our mission to ensure that this cloud infrastructure aligns with best practices and regulatory standards, safeguarding not only the confidentiality of patient data but also the continuous availability and integrity of healthcare services.

Our investigation has revealed several security issues within the infrastructure, including weak access controls, unencrypted sensitive data, vulnerable virtual machines, inadequate network security, lack of robust logging and monitoring, and insufficient disaster recovery planning. Each of these issues poses a significant risk to the confidentiality, integrity, and availability of healthcare services and patient data.

This report aims to provide an in-depth analysis of these vulnerabilities and recommend strategies for addressing them to ensure the security and compliance of the cloud infrastructure in use by the healthcare company. By addressing these concerns, we will help the organization fortify its cloud environment against potential threats and safeguard the privacy and well-being of its patients.

Section 2: General Vulnerability Assessment

Within the context of cloud infrastructure, we have harnessed the power of the Prowler tool[1] to illuminate vulnerabilities across the domains of Weak Access Controls, Lack of Logging and Monitoring, and other Security tools.

Our focus within this assessment is multi-faceted. Firstly, we delve into the intricate world of Identity and Access Management (IAM) policies. We scrutinize their configurations, identifying vulnerabilities, and seeking to rectify excessive permissions and unauthorized access to critical resources. Secondly, we turn our attention to the ominous specter of weak access controls, which can potentially serve as gateways for unauthorized entry to our most critical assets. Third, we look at the deployment code which sets up the cloud infrastructure as it is filled with insecure configurations and insecure passwords. Lastly, we look at some of the AWS services that perform automated vulnerability scans on the infrastructure that are disabled in the current setup.

1. Admin Role and IT Department IAM Policy

The IAM policy designated for the organization's Cloud Admin and IT department(developer) role, with its broad allowance of all AWS actions and access to all resources, is fundamentally misconfigured, introducing excessive permissions and a profound risk of unauthorized access to critical resources. This policy, in effect, grants unrestricted access to the entire AWS environment, thereby circumventing vital access controls and security measures. The severity of this misconfiguration is extraordinarily high, as it leaves the entirety of our cloud infrastructure susceptible to potential breaches, data exposure, and service disruptions. The implications for business risk and continuity are dire, as it places sensitive patient data, telemedicine consultations, and vital healthcare applications at grave risk. The threat it poses to the organization is monumental, with the potential for data breaches, unauthorized modifications, and the impairment of essential healthcare services.

To remediate this critically misconfigured IAM policy, it is imperative to adhere to the principle of least privilege. A more secure and focused IAM policy should be crafted, granting only the essential permissions necessary for the respective role's specific duties. A more secure and focused IAM policy should be crafted, granting only the necessary permissions for the tasks at hand. AWS provides robust tools for managing IAM policies and permissions, allowing for fine-grained control over access.

For example, the IT department requires full access only to the frontend and backend instances for management purposes. Thus, instead of giving complete access to all resources, the IT department policy should be modified to give access only to specific services like S3, RDS, and EC2 instances.

2. Insecure Infrastructure Deployment Code

The YML code provided for deploying a healthcare company's cloud infrastructure on AWS raises several security concerns and vulnerabilities that can pose significant threats to the organization's sensitive patient data. One of the primary concerns is the overly permissive security group configurations. The "PublicSecurityGroup" and "InstanceSecurityGroup" allow open access to ports 22 (SSH) and 80 (HTTP) from any source, which can lead to unauthorized access and potential breaches. Furthermore, the RDS database security group "RDSSecurityGroup" allows unrestricted access to port 3306 (MySQL), which is only limited to the private subnets but still poses a risk. Inadequate security measures, such as weak password policies and a lack of multi-factor authentication, can expose the MySQL database to brute force attacks or unauthorized access. These vulnerabilities can result in data breaches, unauthorized data access, and potential legal and regulatory consequences, affecting the organization's reputation, business continuity, and compliance with healthcare data protection regulations like HIPAA.

To enhance the security of this deployment, several modifications and best practices can be implemented. First, the security group configurations should be tightened to restrict access to only trusted IP addresses and limit open ports to necessary services. For example, restricting SSH access to specific IP ranges and limiting access to the MySQL database to trusted internal services can mitigate risks. Stronger password policies and implementing multi-factor authentication for both SSH and database access can enhance security. Additionally, regular security patching and monitoring of the AWS resources can help detect and mitigate potential vulnerabilities in the infrastructure. Moreover, encrypting sensitive data at rest and in transit, along with implementing audit trails and access logs, can further enhance the security posture and aid in compliance with healthcare data protection regulations. Regular security audits and penetration testing can identify potential weaknesses and provide actionable insights to improve security.

3. Vulnerability: Absence of Security Audit Role

Severity: Low
Business Risk and Continuity: Limited
Threat: Unauthorized Security Access

The SecurityAudit policy has not been attached to any IAM role, raising concerns about the clear separation of duties and access permissions within our organization. The severity of this issue is moderate, as it potentially blurs the lines of responsibility and expertise, allowing unauthorized individuals access to security-related activities. While the business risk and continuity implications are limited, the lack of a dedicated Security Audit role increases the risk of unauthorized security access, which, if not promptly addressed, could lead to compliance violations and data breaches.

To remedy this vulnerability, it is essential to create an IAM role specifically tailored for conducting security audits. This role should be equipped with the necessary SecurityAudit policy, ensuring that only authorized and knowledgeable individuals are granted access to security-related activities. There should be a specific user from the IT department with the expertise to perform security audits assigned with the SecurityAudit policy. Detailed guidance on creating such roles can be found in the IAM Security Audit User Guide [2] documentation webpage.

4. Vulnerability: Weak IAM Password Policy

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Password Security Weakness

Our meticulous examination of the AWS Identity and Access Management (IAM) configuration has unearthed a vulnerability related to the absence of a robust IAM password policy. This absence raises concerns about password security within our organization. The risk associated with this vulnerability is moderate, as it exposes us to potential password security weaknesses, which could lead to unauthorized access or breaches. While the immediate business risk and continuity implications are limited, the threat to our infrastructure and organization cannot be understated.

To mitigate this vulnerability, it is essential to establish a stringent IAM password policy. This policy should encompass critical elements such as password complexity requirements, including the use of different character sets, uppercase

letters, numbers, and non-alphanumeric characters. Additionally, we should ensure that the password expiration period is set to 90 days or less, further enhancing the security of our IAM accounts. Detailed guidance on creating password policies can be found in the password user guide [3] AWS documentation page.

5. Vulnerability: Disabled IAM Access Analyzer

Severity: Low
Business Risk and Continuity: Minimal
Threat: Unidentified Unintended Access

IAM Access Analyzer is a valuable security tool designed to identify resources shared with external entities, helping to uncover unintended access to our resources and data. The severity of this vulnerability is low, but it introduces the risk of not identifying potential security breaches or unauthorized access. While the immediate business risk and continuity implications are minimal, it is essential to address this vulnerability to maintain a robust security posture.

To remedy this situation, it is crucial to enable IAM Access Analyzer for all accounts. This security feature employs automated reasoning to identify access paths allowed by resource policies, contributing to a proactive security stance. By enabling IAM Access Analyzer, we can promptly identify and rectify any unintended access to our resources, reducing the threat of data breaches and unauthorized access. Further details on enabling the IAM Access Analyzer can be found in the AWS documentation of the IAM User Guide [4].

6. Vulnerability: Inactive AWS Inspector2

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Reduced Security Vulnerability Detection

Currently, AWS Inspector2 is not enabled, which significantly limits our capability to identify and address security vulnerabilities in our AWS resources. The severity of this vulnerability is moderate, as it poses a risk of missing critical security vulnerabilities, potentially leading to unauthorized access, data breaches, or other security incidents. Although the immediate business risk and continuity impact is limited, the threat of reduced security vulnerability detection is a concern.

To mitigate this vulnerability, it is strongly recommended to enable AWS Inspector2. AWS Inspector is a valuable tool for identifying security vulnerabilities in AWS resources and provides essential insights for enhancing security. Detailed instructions for enabling Inspector2 can be found in the AWS documentation at AWS Inspector [5]. By activating AWS Inspector2, we can proactively detect and address security vulnerabilities, bolster our security posture, and reduce the risk of unauthorized access, data breaches, and other security incidents.

7. Vulnerability: Inactive AWS IAM MFA

Severity: Medium
Business Risk and Continuity: Minimal
Threat: Compromised Authentication and Unauthorized Access

The lack of Multi-Factor Authentication (MFA) for AWS Identity and Access Management (IAM) users presents a substantial security vulnerability within our cloud infrastructure. This issue has a high severity rating due to the elevated risk of unauthorized access to AWS resources. The absence of MFA could lead to compromised authentication credentials, making it considerably easier for malicious actors to gain control of an IAM user's account. Given that IAM users often have access to sensitive data and critical systems, the risk to business continuity and security is significant. Without MFA, we are exposed to potential breaches that could disrupt operations and result in data loss or theft.

To mitigate this vulnerability and strengthen our security posture, it is crucial to enforce MFA for all IAM users. This measure adds a layer of security on top of the traditional username and password, significantly reducing the risk of unauthorized access. We should enforce MFA across the board and provide training to ensure all users understand how to use it effectively. The AWS Management Console offers a straightforward process for enabling MFA, and detailed guidance is available on the AWS documentation page IAM MFA [6].

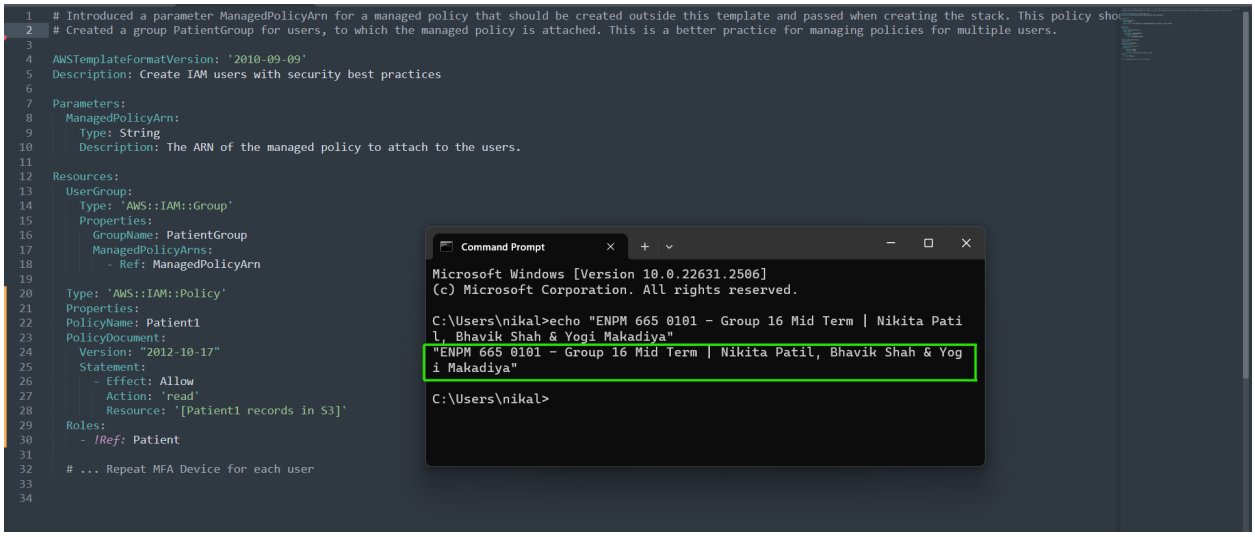


Figure 2: Sample secure IAM policy for Patient 1 in YML format.

In conclusion, the Vulnerability Assessments have shed light on various vulnerabilities within our AWS environment, each carrying its own unique set of risks. Weak IAM policies have the potential to grant excessive permissions and unauthorized access, putting critical resources at risk. Insufficient access controls and insecure deployment of YML code can collectively create gaps in our security posture, making it challenging to detect and respond to security incidents effectively.

Section 3: Data Security Assessment

In this section, we delve into the critical realm of Data Security Assessment within the company's AWS infrastructure. Leveraging the insights provided by the Prowler tool [1], we have meticulously identified vulnerabilities related to data security across a spectrum of AWS services, including S3 and RDS. The company places paramount importance on the cloud for storing and managing sensitive patient data, enabling telemedicine consultations, and hosting vital healthcare applications. At the heart of this ecosystem lies an application designed to handle and safeguard sensitive patient information, encompassing medical records, personal details, and billing information. The integrity and confidentiality of this data are not merely a compliance requirement but a moral obligation, as it plays a pivotal role in patient care and trust. In this context, our Data Security Assessment takes center stage, as we embark on the mission to fortify our defenses, uphold regulatory compliance, and ensure the sanctity of the data that is fundamental to our healthcare operations.

1. Vulnerability: S3 Bucket MFA Delete Not Enabled

   Severity: Medium
   Business Risk and Continuity: Moderate
   Threat: Unauthorized Deletion and Unauthorized Access

   The S3 bucket deployed by the company using the YML code has MFA Delete disabled. This configuration oversight introduces a medium-severity risk to our organization's data security. MFA Delete, when not enabled, allows for the potential unauthorized deletion of objects and versions within the bucket. This not only poses a threat to data integrity and availability but also increases the risk of unauthorized access to sensitive information. While the threat is not immediate, the moderate business risk and continuity concerns make it crucial to address this vulnerability promptly.

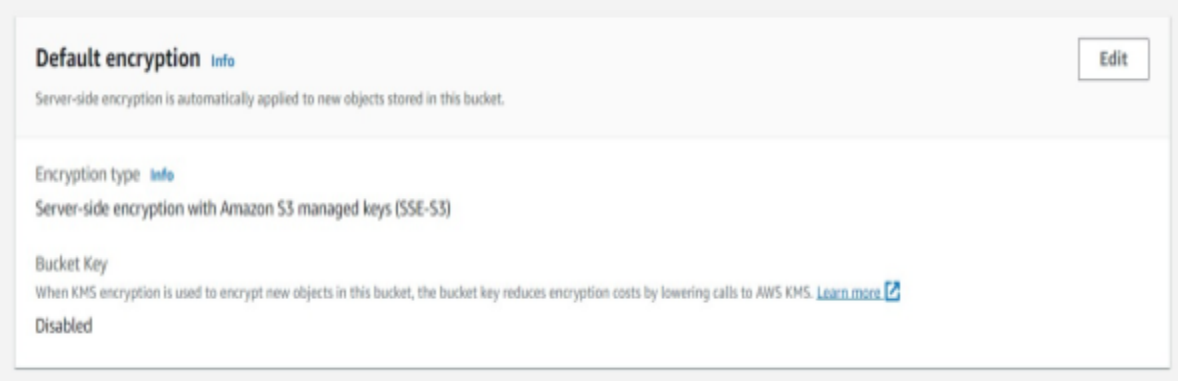
   To mitigate this vulnerability, it is strongly recommended that we enable MFA Delete for the S3 bucket. Enabling MFA Delete adds a layer of security by requiring multi-factor authentication when changing the version state of the bucket or deleting object versions. Detailed guidance on how to implement MFA Delete for an S3 bucket can be found in the MFA Delete [7] documentation.

2. Vulnerability: S3 Buckets Missing KMS Encryption

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Data-at-Rest Vulnerability

The S3 bucket does not have Key Management Service (KMS) encryption configured. This oversight introduces a medium-severity risk to our data-at-rest security. Amazon S3 KMS encryption provides a mechanism to ensure that data stored in the S3 bucket is encrypted at rest, which is crucial for safeguarding sensitive information. The absence of this encryption leaves our data vulnerable to unauthorized access or data breaches, compromising both data integrity and security. Although the threat is not immediate, the moderate business risk and continuity concerns make it imperative to address this vulnerability promptly.

To mitigate this vulnerability, it is strongly recommended that we enable encryption at rest using KMS for our S3 buckets. This can be achieved by configuring the appropriate KMS key for the bucket, which adds a layer of security to protect data-at-rest. Detailed guidance on implementing KMS encryption for S3 buckets can be found in the S3 KMS Encryption [8] AWS documentation.



Screenshot 3: KMS Encryption Disabled

3. Vulnerability: S3 Bucket Object Lock Disabled

Severity: Low
Business Risk and Continuity: Limited
Threat: Data Deletion and Ransomware Attacks

The S3 bucket has Object Lock disabled. This configuration oversight introduces a low severity risk to our data security. Object Lock is designed to store objects using a Write-Once-Read-Many (WORM) model, which prevents objects from being deleted or overwritten for a fixed duration or indefinitely. By disabling Object Lock, we expose our data to the risk of accidental or malicious data deletions. Furthermore, in the era of increasing ransomware attacks, the absence of Object Lock heightens the threat of data compromise. The threat is not immediate, but it introduces limited business risk and could affect the continuity and performance of our infrastructure and services.

To mitigate this vulnerability, it is strongly recommended that we enable the Object Lock feature for our Amazon S3 buckets. Object Lock serves as a crucial safeguard against unwanted data deletions and provides a defense mechanism against ransomware attacks. Detailed guidance on configuring and using Object Lock for S3 buckets can be found in the AWS documentation of S3 Object Lock [9].

4. Vulnerability: S3 Bucket Versioning Disabled

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Data Loss and Recovery Challenges

This configuration oversight introduces a medium-severity risk to our data integrity and recovery capabilities. With versioning disabled, the bucket lacks the crucial ability to maintain and recover from unintended user actions or application failures. This exposes our data to potential loss and complicates the recovery process in the face of data corruption or unintentional deletions. The threat is not immediate, but the moderate business risk and continuity concerns make it crucial to address this vulnerability promptly.

To mitigate this vulnerability, it is strongly recommended that we enable Object Versioning for our Amazon S3 buckets. Object Versioning allows us to maintain a complete history of all object versions within the bucket, enabling us to recover from accidental and malicious data changes. Detailed guidance on configuring

and using Object Versioning for S3 buckets can be found in the AWS documentation for Bucket Versioning [10].

5. Vulnerability: S3 Bucket Missing Secure Transport Policy

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Data Exposure and Network Vulnerability

This configuration oversight introduces a medium-severity risk to our data security and network communications. Without HTTPS enforcement on the bucket policy, communication between clients and the S3 bucket can occur over unencrypted HTTP, potentially exposing sensitive information to eavesdropping during transmission over the network or the internet. While the threat is not immediate, the moderate business risk and continuity concerns necessitate prompt action to address this vulnerability.

To mitigate this vulnerability, it is strongly recommended that we enforce encryption in transit for our S3 buckets. By enabling secure transport policies and enforcing HTTPS, we can protect data during transmission and safeguard it from potential exposure. Detailed guidance on configuring secure transport policies for S3 buckets can be found on the Secure Transport Policy [11] AWS documentation page.

6. Vulnerability: Unencrypted RDS Instance Storage

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Data-at-Rest Vulnerability

This configuration oversight introduces a medium-severity risk to our data protection and privacy. Without encryption, sensitive information stored within the RDS instance is vulnerable to unauthorized access or data breaches. The threat is not immediate, but the moderate business risk and continuity concerns make it crucial to address this vulnerability promptly.

To mitigate this vulnerability, it is strongly recommended that we enable encryption for the storage of our RDS instances. Enabling encryption ensures the confidentiality and integrity of data-at-rest, providing additional management and privacy benefits. Detailed guidance on enabling encryption for RDS instances can be found in the documentation of RDS Encryption [12].

7. Vulnerability: RDS Instance Deletion Protection Not Enabled

Severity: Low
Business Risk and Continuity: Limited
Threat: Unintended Data Loss

This configuration oversight introduces a low severity risk to our data integrity. Deletion protection is a vital safeguard that prevents unintended deletion of critical RDS instances. While the threat is not immediate, the limited business risk and continuity concerns make it important to address this vulnerability to reduce the risk of inadvertent data loss.

To mitigate this vulnerability, it is strongly recommended that we enable deletion protection for our RDS instances. Deletion protection ensures that critical database instances are shielded from unintentional deletion, providing an added layer of security. Guidance for enabling deletion protection for RDS instances can be found in the AWS delete protection [13] document.

8. Vulnerability: RDS Instance Lacks Multi-AZ Configuration

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Availability Zone Failure Resilience

This configuration oversight introduces a medium-severity risk to our database availability and continuity. Without Multi-AZ deployment, our database is vulnerable to potential downtime in the event of an Availability Zone-specific failure. This can result in service disruptions and data unavailability. While the threat is not immediate, the moderate business risk and continuity concerns make it essential to address this vulnerability promptly.

To mitigate this vulnerability, it is strongly recommended that we enable Multi-AZ deployment for our production databases. Multi-AZ deployment provides automatic failover to a standby Availability Zone in the event of a primary Availability Zone failure, ensuring high availability and resilience. Detailed guidance on enabling Multi-AZ deployment for RDS instances can be found in the AWS documentation of RDS Multi AZ Configuration [14].

9. Vulnerability: Macie Disabled

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Reduced Data Security and Privacy Protection

The deactivation of Amazon Macie represents a critical security oversight in the cloud infrastructure of a healthcare organization handling sensitive personal health information (PHI). Without Macie's advanced data security features, which leverage machine learning and pattern matching, the company is exposed to significant risks. The risk includes potential data breaches, unauthorized access, and the compromise of data integrity. The high severity of this vulnerability is underscored by the substantial business and continuity risks it poses, threatening compliance with stringent healthcare regulations and the organization's ability to safeguard patient data.

Remediation of this vulnerability necessitates the immediate activation of Amazon Macie. By enabling Macie, the organization can benefit from its robust capabilities in identifying, classifying, and securing sensitive data within Amazon S3. Configuring Macie to monitor for PHI and PII is critical for maintaining compliance with healthcare regulations such as HIPAA. The comprehensive data security measures provided by Macie are crucial for the protection of sensitive information and help mitigate the risk of costly data breaches. Detailed guidance for activating and configuring Macie [15] is available in the AWS documentation.

Through the above comprehensive Data Security Assessment, we have examined the vital aspects of data protection within our AWS environment, with a focus on Amazon S3 and RDS services. The risks associated with unencrypted data, data-at-rest vulnerabilities, unauthorized deletions, data exposure, data loss, and network vulnerabilities have been brought to light. These vulnerabilities underscore the paramount importance of implementing robust data security measures to safeguard sensitive information from potential threats. These vulnerabilities also makes the infrastructure non-compliant to PCI DSS and HIPAA standards.

The recommendations provided for secure data storage and encryption stand as essential steps to mitigate these risks. Enabling encryption for data-at-rest, both in S3 buckets and RDS instances, is critical to maintaining the confidentiality and integrity of our sensitive data. Additionally, configuring versioning, object lock, secure transport policies, and enabling enhanced monitoring further fortifies our data security posture.

Section 4: VM Vulnerability Assessment

The organization has decided to host their healthcare applications and databases related to the application using a VM that is set up on Amazon Web Services (AWS). Presently, they are hosting an EC2 machine with a network open to the entire Internet and with port 22 open for SSH connections. Using Prowler [1], we performed a vulnerability assessment identifying the vulnerabilities and suggested remediations and patches as per the best practices and configurations suggested by AWS.

1. Vulnerability: EC2 Instances with Public IP (Inactive WAF)

Severity: Medium
Business Risk and Continuity: Elevated
Threat: Increased Attack Surface

This exposure increases the attack surface and, consequently, raises the risk of a potential compromise. The presence of a publicly accessible EC2 instance, in this context, introduces a significant security threat to our organization's cloud environment. The severity of this vulnerability is of paramount concern, as it exposes a critical component of our infrastructure directly to the internet, rendering it susceptible to a wide range of potential threats, including unauthorized access, data breaches, and malicious activity. The business risk associated with this vulnerability is substantial, as a successful compromise of this EC2 instance could result in data breaches, service disruptions, and reputational damage. Furthermore, it poses a significant threat to the overall integrity of our cloud infrastructure and, by extension, the organization itself. Addressing this vulnerability is imperative to mitigate these risks effectively.

To mitigate this vulnerability and ensure the security of our cloud environment, we strongly recommend the implementation of an Application Load Balancer (ALB) in conjunction with the AWS Web Application Firewall (WAF). The ALB will act as a protective intermediary layer, distributing traffic to EC2 instances while hiding their public IP addresses. By configuring WAF Access Control Lists (ACLs), we can filter and monitor incoming traffic for potential threats and malicious activity. This approach will significantly enhance the security of our EC2 instances, safeguarding them from direct exposure to the internet and thereby reducing the risk of compromise. To implement this solution, we advise following the guidance provided in AWS documentation for ALB and AWS Web Application Firewall, which can be found at AWS WAF & ALB [16].

2. Vulnerability: EBS Default Encryption Deactivation

Severity: Medium
Business Risk and Continuity: Elevated
Threat: Data Exposure

It has been identified that the default encryption for Amazon Elastic Block Store (EBS) volumes is not activated. This oversight increases the risk to our sensitive data at rest. EBS default encryption, when not enabled, leaves our stored information susceptible to unauthorized access in the event of a breach or compromise. This vulnerability poses a moderate to high level of severity, as it directly affects the confidentiality and integrity of our data. The potential threat to our organization includes data exposure, loss of sensitive information, and reputational damage, which could have lasting consequences.

To address this vulnerability and enhance our data security, we must enable EBS encryption. Additionally, whenever possible, we should utilize a Customer Master Key (CMK) to provide enhanced management and privacy benefits. AWS offers comprehensive guidance on how to enable automatic EBS encryption, which can be found at EBS Automatic Encryption.

3. Vulnerability: Missing EBS Snapshots for the instance

Severity: Low
Business Risk and Continuity: Limited
Threat: Data Recovery Challenges

We discovered that there are no recent snapshots available for the EBS volume. This vulnerability introduces a low-severity risk, as it pertains to our ability to recover data in the event of an incident or failure. The absence of snapshots hinders our ability to perform point-in-time recovery, and it poses a limited business risk as it could result in data loss if the EBS volume experiences issues. While the threat is not immediate, it is essential to maintain reliable data backup strategies to ensure business continuity.

To address this vulnerability, it is strongly recommended that we create regular point-in-time EBS snapshots for our volumes. Regular snapshots taken weekly can serve as a crucial component of our data recovery strategy and provide a reliable backup for geographical expansion and disaster recovery purposes. AWS provides detailed guidance on how to create and manage EBS snapshots, which can be found at EBS Snapshot [17].

4. Vulnerability: EC2 Instance Metadata Service Version 2 (IMDSv2) Disabled or Not Required

Severity: Medium
Business Risk and Continuity: Moderate
Threat: SSRF and Misconfiguration Vulnerabilities

This configuration oversight introduces a medium-severity risk to our infrastructure and organization. IMDSv2 is designed to protect against misconfiguration and Server-Side Request Forgery (SSRF) vulnerabilities, offering enhanced security and risk mitigation. The use of IMDSv1 leaves our EC2 instances susceptible to potential SSRF attacks and misconfiguration issues, which could compromise system integrity and potentially lead to unauthorized access or data exposure. While the threat is not immediate, the moderate business risk and continuity concerns make it imperative that we address this vulnerability promptly.

To mitigate this vulnerability, it is strongly recommended that we ensure IMDSv2 is enabled and required for our EC2 instances. If IMDS is not required for specific instances, it can be turned off. Using the AWS Command Line Interface (aws-cli), we can enforce the use of IMDSv2 for our instances to enhance their security. Detailed guidance on configuring instance metadata options can be found in the IMDSv2 [18] AWS documentation.

5. Vulnerability: EC2 Instances Not Managed by Systems Manager

Severity: Low
Business Risk and Continuity: Limited
Threat: Reduced Compliance and Monitoring

While this vulnerability presents a low severity risk, it impacts our ability to ensure compliance with common best practices and maintain an efficient monitoring strategy. EC2 instances not managed by the Systems Manager may be subject to misconfigurations, drift from desired states, and a lack of automated maintenance, which could lead to operational inefficiencies or potential security vulnerabilities. The threat is not immediate, but it introduces limited business risk and could affect the continuity and performance of our infrastructure and services.

To address this vulnerability, it is strongly recommended that we verify and apply Systems Manager prerequisites and integrate EC2 instances into our Systems Manager management framework. AWS Systems Manager provides a comprehensive solution for managing, monitoring, and automating our EC2 instances, ensuring they comply with best practices and reducing operational overhead. Detailed guidance on managing EC2 instances using System Manager [19] can be found in the AWS documentation.

6. Vulnerability: EC2 Instance Missing IAM Instance Profile Role

Severity: Medium
Business Risk and Continuity: Moderate
Threat: Unauthorized Access and Credential Compromise

The EC2 instance is not associated with an Instance Profile Role. This configuration oversight introduces a medium-severity risk to our infrastructure and organization. The absence of IAM roles for EC2 instances can lead to AWS access being managed through the encoding of AWS keys into API calls, potentially exposing sensitive credentials within the instance. This not only increases the risk of unauthorized access but also poses a significant threat if credentials are compromised. If credentials fall into the wrong hands, they could be misused from outside of our AWS account. While the threat is not immediate, the moderate business risk and continuity concerns make it crucial to address this vulnerability promptly.

To mitigate this vulnerability, we must create and attach IAM instance roles to EC2 instances where necessary. By associating the instance with an IAM role, we significantly reduce the risks associated with credential exposure and unauthorized access, as the role enforces appropriate permissions policies for the required access. Detailed guidance on how to create and use IAM roles for EC2 [20] instances can be found in the AWS documentation.

In our detailed examination of VM Vulnerabilities within our AWS infrastructure, we have brought to light a multitude of potential risks, ranging from increased attack surface to unauthorized access and credential compromise. These vulnerabilities emphasize the critical importance of proactively addressing security gaps within our EC2 instances. The recommendations we have provided, including patching, enabling encryption, creating EBS snapshots, implementing Systems Manager, and enforcing stringent security controls, serve as a strategic roadmap for mitigating these risks effectively.

As custodians of a secure and resilient cloud environment, we must continue to diligently apply these recommendations and stay vigilant in our efforts to secure our EC2 instances. By doing so, we not only reduce the risk of data breaches, unauthorized access, and system vulnerabilities but also reinforce our commitment to maintaining a robust and secure cloud infrastructure.

Section 5: Network Security Assessment

In this section, we delve into a critical aspect of our cloud infrastructure – network security. Utilizing the insights from the Prowler tool [1], we have identified vulnerabilities that demand our immediate attention. Our cloud environment boasts a multitude of virtual networks and subnets, each playing a distinct role in serving the needs of various departments and services within our organization. The misconfigurations and gaps in network security place our sensitive resources at risk, potentially exposing them to unauthorized access and other security threats. In this section, we aim to comprehensively address these network security vulnerabilities. By doing so, we can fortify our cloud infrastructure, ensuring that our networks and subnets are both robust and secure, and safeguarding the integrity and confidentiality of our sensitive data.

1. Vulnerability: Default Security Groups Allowing Unrestricted Traffic

Severity: High
Business Risk and Continuity: Elevated
Threat: Unauthorized Access and Security Breaches

The default security group rules have been configured to allow traffic, which introduces a high-severity risk to our infrastructure and organization. Allowing unrestricted traffic within default security groups opens a gateway for unauthorized users or malware with VPC access to scan for well-known and sensitive ports, potentially gaining access to our instances. This misconfiguration poses an immediate threat to the security of our AWS resources and data. Unauthorized access could lead to data breaches, service disruptions, or even complete security breaches, all of which present an elevated business risk and continuity concern. We must address this vulnerability promptly to safeguard our AWS infrastructure.

To mitigate this vulnerability, we must adopt a Zero Trust approach and implement strict security practices to remediate overly permissive default security groups. Recommended best practices involve narrowing the definition of minimum required ports and restricting access based on legitimate use cases. AWS provides comprehensive guidance on securing security groups and Network ACLs, which can be found in the AWS documentation page for EKS Security Groups [21].

2. Vulnerability: EC2/VPC Open to Ingress from 0.0.0.0/0 on All Ports, Including RDP and SSH

Severity: High
Business Risk and Continuity: Elevated
Threat: Unauthorized Access and Security Breaches

In our recent evaluation of our AWS Virtual Private Cloud (VPC) security, a critical vulnerability has been uncovered concerning Network Access Control Lists (NACLs). Specifically, Network ACL has been configured to allow ingress from 0.0.0.0/0 (the entire internet) to all ports, including RDP (port 3389) and SSH (port 22). This misconfiguration introduces a high-severity risk to our infrastructure and organization. Allowing unrestricted access to sensitive ports like RDP and SSH creates a direct threat to the security of our instances and data. Unauthorized users or malware with VPC access could scan for these well-known ports and potentially gain access to instances, leading to unauthorized data access, service disruption, or even complete security breaches. The threat is immediate, and the elevated business risk and continuity concerns necessitate prompt attention to remediate this vulnerability.

To address this vulnerability, we must adopt a zero-trust approach and implement strict security practices to remediate overly permissive Network ACL rules. Recommended best practices involve narrowing the definition of minimum required ports and restricting access based on legitimate use cases. AWS provides comprehensive guidance on securing security groups and Network ACLs, which can be found in the AWS documentation for Security Groups [22].

3. Vulnerability: Public IP Assignment in VPC Subnet

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Unauthorized Internet Exposure

The company's infrastructure subnet is currently configured to assign public IP addresses to instances automatically upon launch. While this may seem convenient, it poses a moderate risk to our infrastructure, as it inadvertently exposes instances within this subnet to the internet. Such unauthorized internet exposure can lead to security breaches, unauthorized access, and other potential threats. Though the immediate business risk and continuity impact is limited, the threat of unauthorized internet exposure necessitates attention.

To mitigate this vulnerability, it is essential to modify the configuration of the VPC subnet and ensure it does not allow automatic public IP assignment. Detailed guidance on how to achieve this can be found in the AWS documentation under subnet assignment [23].

4. Vulnerability: Lack of VPCs in Multiple Regions

Severity: Moderate
Business Risk and Continuity: Moderate
Threat: Region-Specific Failures

Currently, the AWS environment hosts VPCs in only one region, which poses a moderate risk to our infrastructure. In the event of region-specific failures or outages, we may experience service disruptions or downtime in critical applications. This vulnerability affects business continuity, as our reliance on a single region can lead to significant operational challenges.

To address this vulnerability and improve the resiliency of our AWS environment, it is crucial to create VPCs in multiple regions. This approach aligns with AWS best practices and helps mitigate the risk of region-specific failures. Detailed guidance on implementing this recommendation can be found in the AWS documentation under VPC [24].

5. Vulnerability: Lack of Network Firewall in VPC

Severity: High
Business Risk and Continuity: Elevated
Threat: Traffic Monitoring and Unauthorized Access

Without an effective network firewall in place, monitoring and controlling traffic within the VPC becomes challenging. This situation can lead to difficulties in detecting and preventing attacks or unauthorized access to critical resources. The absence of a Network Firewall in the VPC presents a security gap that could potentially be exploited by threat actors, impacting the integrity and security of our AWS environment.

To address this vulnerability and enhance the security and control of our VPC, it is recommended to ensure that all VPCs have a Network Firewall enabled. AWS Network Firewall plays a crucial role in safeguarding our VPCs by monitoring and

filtering traffic effectively. Detailed guidance on configuring Network Firewall for VPCs can be found in the AWS documentation under Network Firewall [25].

The Network Security Assessment of our cloud infrastructure has brought to light a range of vulnerabilities that, if left unaddressed, could pose significant risks to our digital assets. The vulnerabilities identified include default security groups permitting unrestricted traffic, permissive configurations of EC2 instances and VPCs that are open to ingress from 0.0.0.0/0 on all ports, including sensitive ones such as RDP and SSH, automatic public IP assignment within VPC subnets, the absence of VPC presence in multiple regions, and the lack of network firewalls within VPCs. These findings underscore the pressing need for comprehensive network security enhancements.

Addressing these vulnerabilities is not just a matter of best practices; it is a fundamental requirement to ensure the safety, privacy, and integrity of our critical data and resources. Fortunately, the assessment has also provided us with valuable recommendations for remediation and patches. By heeding these recommendations and implementing the necessary changes, we can effectively fortify our network security posture.

Section 6: Logging and Monitoring Assessment

Our Logging and Monitoring Assessment, conducted with the assistance of the Prowler tool [1], has revealed vulnerabilities and shortcomings within our logging and monitoring capabilities. These findings have unearthed several inactive services that are essential for monitoring malicious activities, as well as for tracking any activities related to policy changes and infrastructure. The absence of robust logging and monitoring mechanisms in our infrastructure poses a significant challenge to our ability to detect and respond effectively to security incidents.

In this section, we embark on a journey to comprehensively analyze our logging and monitoring capabilities. Our goal is to identify and address these weaknesses, and, through a set of recommendations, to propose improvements that will empower us to strengthen our ability to detect and respond to security incidents proactively. By enhancing our logging and monitoring, we reaffirm our commitment to the security, privacy, and compliance requirements of our patients and the integrity of our healthcare operations.

1. Vulnerability: Inactive GuardDuty

   Severity: Moderate
   Business Risk and Continuity: Limited
   Threat: Reduced Security Monitoring

   Currently, GuardDuty is not enabled, which reduces our ability to maintain continuous security monitoring. The severity of this vulnerability is moderate, as it limits our capacity to analyze and process critical data sources for security threat detection. While the immediate business risk and continuity implications are limited, the reduced security monitoring poses a threat to our infrastructure and organization by potentially missing early signs of security threats.

   To address this vulnerability, it is strongly recommended to enable Amazon GuardDuty. GuardDuty is a valuable security monitoring service that can analyze and process various data sources to detect and respond to security threats effectively. Detailed instructions for setting up GuardDuty [26] can be found in the AWS documentation.

2. Vulnerability: Absence of CloudWatch Log Metric Filters and Alarms

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Limited Visibility into Security Events

Our thorough review of the AWS environment has identified a vulnerability related to the lack of log metric filters and alarms. Specifically, there are no CloudWatch log groups with associated metric filters and alarms configured for various critical events, including AWS Config changes, AWS Management Console authentication failures, AWS Organizations changes, Network Access Control Lists (NACL) modifications, network gateway adjustments, CloudTrail configuration changes, disabling or scheduled deletion of customer-created KMS CMKs, IAM policy changes, S3 bucket policy changes, security group adjustments, unauthorized API calls, and VPC changes. The severity of this vulnerability is moderate, as it diminishes our ability to monitor and respond to unauthorized requests and security-related events. While the immediate business risk and continuity implications are limited, the reduced visibility into potential security threats necessitates prompt remediation.

To mitigate this vulnerability, it is crucial to establish log metric filters and alarms for the mentioned critical events. These filters and alarms help us monitor unauthorized API calls, reveal application errors, and reduce the time required to detect and respond to malicious activity. It is recommended to establish metric filters and alarms for unauthorized requests, as outlined in the AWS documentation of CloudWatch [27].

3. Vulnerability: Absence of Object-level S3 Logging in CloudTrail

Severity: Moderate
Business Risk and Continuity: Limited
Threat: Reduced Monitoring and Threat Analysis

This deficiency means that no CloudTrail trails are configured to record all S3 object-level API operations. The severity of this vulnerability is moderate, as it hampers our ability to monitor service usage and conduct effective threat analysis. This also makes our application open to PCI DSS non-compliance as we cannot track whether any change or deletion was made to customer's payment information. While the immediate business risk and continuity implications are limited, the threat of reduced monitoring and the potential inability to detect and respond to security incidents necessitates a prompt remedy.

To address this vulnerability, it is imperative to enable object-level logging for S3 in CloudTrail. By doing so, we can proactively monitor all S3 object-level API operations, providing valuable insights into service usage and security-related activities. Additionally, we should create an S3 lifecycle policy and define use cases, metrics, and automated responses where applicable. Detailed instructions on enabling CloudTrail logging for S3 [28] can be found in the AWS documentation.

4. Vulnerability: AWS Security Hub disabled

Severity: Moderate
Business Risk and Continuity: Moderate
Threat: Limited Security Monitoring and Alerting

AWS Security Hub offers a comprehensive solution for obtaining a unified view of security alerts and the overall security posture across our AWS accounts. The absence of an active AWS Security Hub leaves us with limited visibility into potential security threats and vulnerabilities, which can lead to difficulties in monitoring and responding to security incidents effectively. This vulnerability may expose our organization to risks associated with undetected security issues that could affect the integrity of our AWS infrastructure and critical assets.

To address this vulnerability and strengthen our security posture, it is strongly recommended to enable AWS Security Hub and configure its standard subscriptions. AWS Security Hub is region-specific, meaning it needs to be enabled and configured in each region individually. Detailed guidance on enabling and configuring standard subscriptions in AWS Security Hub can be found in the AWS documentation under Security Hub User Guide [29].

5. Vulnerability: Absence of Enhanced Monitoring in RDS Instance

Severity: Low to Moderate
Business Risk and Continuity: Low to Moderate
Threat: Limited Visibility into OS Metrics

This instance does not have enhanced monitoring enabled, which can result in limited visibility into OS metrics and other crucial performance data. Enhanced monitoring is a valuable feature offered by AWS that allows us to collect a more extensive set of metrics and improve the granularity of data collected from the

RDS instance's operating system. Without enhanced monitoring, we may miss the opportunity to gain deeper insights into the performance and health of the database, potentially leading to difficulties in identifying and addressing issues promptly.

To mitigate this vulnerability and enhance our ability to monitor the RDS instance effectively, we recommend enabling enhanced monitoring. This can be achieved by creating an IAM role and subsequently enabling enhanced monitoring for the RDS instance. Detailed guidance on how to set up enhanced monitoring can be found in the AWS documentation under the RDS User Guide [30].

6. Vulnerability: Lack of Detailed Monitoring in EC2 Instance

Severity: Low to Moderate
Business Risk and Continuity: Low to Moderate
Threat: Limited Visibility into Performance Metrics

Detailed monitoring is not enabled for this instance, which means that we currently lack the benefits of enhanced monitoring and granular insights into its performance metrics. The absence of detailed monitoring may limit our ability to effectively troubleshoot performance issues, track resource utilization, and promptly address any performance-related concerns.

To address this vulnerability and enhance our monitoring capabilities, we recommend enabling detailed monitoring for the EC2 instance. Enabling detailed monitoring can be achieved by following the guidelines provided in the AWS documentation under EC2 Monitoring [31].

7. Vulnerability: Lack of billing dashboard and alerts

Severity: Moderate
Business Risk and Continuity: Moderate
Threat: Limited Visibility for Cost Management

The absence of a billing dashboard and alerts in an AWS (Amazon Web Services) environment represents a significant vulnerability as it hinders an organization's ability to effectively monitor and control its cloud costs. Without these essential tools, it becomes challenging to keep track of resource usage, identify cost

overruns, or detect potential billing anomalies promptly. This lack of visibility can lead to unexpected and skyrocketing expenses, making it difficult to optimize resource allocation and potentially leaving the infrastructure exposed to financial risks, ultimately compromising the financial health and security of the AWS environment.

To remediate the lack of a billing dashboard and alerts in AWS, organizations should promptly enable the AWS Cost Explorer and set up billing alerts in AWS Budgets. The AWS Cost Explorer provides detailed cost and usage reports, allowing users to monitor resource expenses effectively. AWS Budgets allows for the creation of custom budgets and alerts based on cost thresholds, ensuring timely notifications when expenses approach or exceed predefined limits. By implementing these measures, organizations can gain real-time insights into their AWS spending and take proactive steps to control costs, thus preventing financial vulnerabilities and ensuring the security and sustainability of their AWS environment. The documentation to configure billing dashboards and generate alerts is on Bill and Alerts [32].

Our Logging and Monitoring Assessment has illuminated critical vulnerabilities within our cloud infrastructure that are central to ensuring the security, integrity, and compliance of our healthcare operations. The identified vulnerabilities encompass a wide spectrum, from the absence of essential AWS services like GuardDuty and Security Hub to gaps in CloudWatch Log Metric Filters and Alarms, and incomplete logging mechanisms in CloudTrail, RDS, and EC2 instances. These vulnerabilities expose our healthcare environment to potential risks, leaving us vulnerable to malicious activities and rendering our response capabilities inadequate.

The suggested remediations and patches are a comprehensive roadmap for strengthening our logging and monitoring capabilities. By enabling services like GuardDuty and Security Hub, and implementing vital features such as CloudWatch Log Metric Filters and Alarms, object-level S3 logging, enhanced monitoring for RDS instances, and detailed monitoring for EC2 instances, we lay the foundation for a more vigilant and responsive cloud infrastructure. These improvements will equip us to identify and address security incidents in real time, enhancing the protection of sensitive patient data, compliance, and the overall integrity of our healthcare applications.

Section 7: Disaster Recovery Assessment

In the wake of comprehensive assessments, including vulnerability evaluations, data security analysis, VM vulnerability scrutiny, network security assessment, and Logging and Monitoring Assessment, our healthcare company's cloud infrastructure has unveiled a range of vulnerabilities and potential weaknesses. These findings underscore the critical importance of addressing the risks associated with data loss and extended downtime. In this section, we will delve into these risks, detailing their potential impact on the organization, and present a comprehensive backup and disaster recovery plan to safeguard our patient data, ensure business continuity, and maintain the integrity of our critical healthcare applications. Our primary goal is to fortify our cloud infrastructure against any unforeseen disasters or disruptions, ultimately ensuring the continued availability and security of sensitive patient information.

In the above 5 sections, we have established the vulnerabilities present in the infrastructure related to all kinds of threat vectors namely:

- Weak IAM Policies
- Insecure Infrastructure Deployment Code
- Unauthorized Security Access
- Password Security Weakness
- Reduced Security Vulnerability Detection
- Compromised Authentication
- Unauthorized Deletion
- Data-at-Rest Vulnerability
- Data Deletion and Ransomware Attacks
- Data Loss and Recovery Challenges
- Availability Zone Failure Resilience
- Reduced Data Security and Privacy Protection
- Increased Attack Surface
- Data Recovery Challenges
- SSRF and Misconfiguration Vulnerabilities
- Reduced Compliance and Monitoring

We also presented the remediations and patches to secure those vulnerabilities. But that is not enough. Any robust security team while securing the organization also requires a proper disaster and recovery plan to handle any event that might occur due to cyberthreats. To address this avenue, we propose a comprehensive backup and disaster recovery plan tailored to the healthcare company's cloud infrastructure:

1. *Regular Backup and Snapshot Procedures*: Implement automated, regular backups and snapshots for EC2 instances, RDS databases, and S3 data, ensuring data is protected and recoverable.
2. *Data Encryption*: Encrypt data at rest and in transit to protect patient information from unauthorized access.
3. *Access Control and IAM Policies*: Strengthen IAM policies and access controls to restrict permissions, ensuring only authorized users have access to sensitive data.
4. *Infrastructure as Code Security*: Review and secure YML code for infrastructure deployment to mitigate code vulnerabilities and misconfigurations.
5. *Multi-AZ Deployments*: Utilize Multi-AZ deployments for critical components to enhance resilience against availability zone failures.
6. *Data Recovery and Restoration Plan*: Develop and regularly test a data recovery and restoration plan to ensure rapid recovery in case of data loss.
7. *Incident Response Plan*: Create an incident response plan to address potential compromises, including ransomware attacks, with well-defined steps and communication procedures.
8. *Monitoring and Compliance*: Implement continuous monitoring and compliance checks to identify and mitigate threats in real time, ensuring adherence to healthcare regulations.
9. *Redundant Data Centers or Regions*: Establish a secondary data center or cloud region to ensure business continuity and data replication for disaster recovery.
10. *User Training*: Provide training to staff on security best practices, password security, and incident response.
11. *Third-party Disaster Recovery Solutions*: Consider third-party disaster recovery services that specialize in healthcare data protection and recovery for added resilience.

Our comprehensive backup and disaster recovery plan, meticulously tailored to tackle the identified threats, is a strategic response to the imperative of safeguarding the healthcare company's cloud infrastructure. It encompasses a multifaceted approach, spanning data backup and encryption, stringent access controls, infrastructure code security, redundancy measures, and incident response protocols. These provisions are thoughtfully designed to address not only data loss and extended downtime but also the security, privacy, and regulatory concerns inherent in the healthcare industry.

As we move forward, it is vital to recognize that the effectiveness of this plan hinges on its regular testing, continuous monitoring, and a commitment to adapting and evolving in the face of an ever-changing threat landscape. The dedication to maintaining high standards of security, compliance, and data integrity will be instrumental in ensuring the continued delivery of quality healthcare services to patients while upholding the organization's reputation and trustworthiness.

Section 8: References

1) https://github.com/prowler-cloud/prowler
2) https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_job-functions.html#jf_security-auditor
3) https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_passwords_account-policy.html
4) https://docs.aws.amazon.com/IAM/latest/UserGuide/what-is-access-analyzer.html
5) https://docs.aws.amazon.com/inspector/latest/user/what-is-inspector.html
6) https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa_enable.html
7) https://docs.aws.amazon.com/AmazonS3/latest/userguide/MultiFactorAuthenticationDelete.html
8) https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingKMSEncryption.html
9) https://docs.aws.amazon.com/AmazonS3/latest/userguide/object-lock-overview.html
10) https://docs.aws.amazon.com/AmazonS3/latest/dev-retired/Versioning.html
11) https://aws.amazon.com/premiumsupport/knowledge-center/s3-bucket-policy-for-config-rule/
12) https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html
13) https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_DeleteInstance.html
14) https://aws.amazon.com/rds/features/multi-az/
15) https://docs.aws.amazon.com/macie/latest/userguide/macie-setting-up.html
16) https://aws.amazon.com/blogs/aws/aws-web-application-firewall-waf-for-application-load-balancers/
17) https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSSnapshots.html
18) https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/configuring-instance-metadata-service.html#configuring-instance-metadata-options
19) https://docs.aws.amazon.com/systems-manager/latest/userguide/managed_instances.html
20) http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_use_switch-role-ec2.html
21) https://docs.aws.amazon.com/eks/latest/userguide/sec-group-reqs.html
22) https://docs.aws.amazon.com/eks/latest/userguide/sec-group-reqs.html
23) https://docs.aws.amazon.com/config/latest/developerguide/subnet-auto-assign-public-ip-disabled.html

24)https://docs.aws.amazon.com/vpc/latest/userguide/vpc-example-private-subnets-nat.html
25)https://docs.aws.amazon.com/network-firewall/latest/developerguide/vpc-config.html
26)https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_settingup.html
27)https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html
28) https://docs.aws.amazon.com/AmazonS3/latest/userguide/enable-cloudtrail-logging-for-s3.html
29)https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-enable-disable.html
30) https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_Monitoring.OS.html
31) https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-cloudwatch-new.html#enable-detailed-monitoring-instance
32) https://docs.aws.amazon.com/account-billing/